

WE CLAIM:

1. A safety controller comprising:
 - a primary and partner independent controller communicating on a communication bus;
 - a communication interface for receiving safety program information from a user to the primary controller;
 - a transfer program executable on the primary and partner controller to automatically load the safety program information received by the primary controller via the communications bus to the partner controller; and
 - a synchronization program executable by the primary and partner controller to execute the safety program information on the primary and partner controller and compares execution of the safety program information and to enter a safety state when this execution differs.
2. The safety controller of claim 1 wherein the communication interface confirms the existence of the partner controller having the transfer and synchronization program and receives safety program information only when the confirmed partner controller is communicating with the primary controller on the communications bus.
3. The safety controller of claim 1 wherein the safety program information executes to generate outputs to be used to control an external device and wherein the synchronization program compares execution of the safety program information by comparing outputs generated by the primary and partner controller executing the safety program information.
4. The safety controller of claim 3 wherein the safety program information is executed repeatedly and wherein the comparison of the outputs is performed at the conclusion of each repeated execution prior to outputting of the outputs to the external device.
5. The safety controller of claim 1 wherein the safety program information executes to generate values of internal variables and wherein the synchronization

program compares execution of the safety program information by comparing values of internal variables generated by the primary and partner controller executing the safety program information.

6. The safety controller of claim 5 wherein the safety program is executed repeatedly and wherein the comparison is performed at a period greater than the repetition period.
7. The safety controller of claim 1 wherein the transfer program transfers the safety program information from the primary controller to the partner controller and receives an acknowledgement from the partner controller to the primary controller indicating that the transfer was complete and correct.
8. The safety controller of claim 7 wherein the transfer program transfers the state program information in portions and receives an acknowledgement for each portion.
9. The safety controller of claim 1 wherein the primary controller holds an identification value indicating to a user device having safety program information that the primary controller may receive safety program information and wherein the partner controller does not hold the identification value indicating to the user device having safety program information that it may receive safety program information.
10. The safety controller of claim 1 wherein the communication interface also receives standard program information and wherein the safety program information holds an identification value indicating that it is part of a safety application and wherein the transfer program checks for this identification value to automatically load only the safety program information received by the primary controller via the communications bus to the partner controller.
11. The safety controller of claim 1 wherein the primary and partner controllers are contained in independent housings separately attachable to an intercommunication bus.

12. The safety controller of claim 11 wherein the second housing holds fewer components than the first housing to provide limited functionality.
13. The safety controller of claim 12 wherein the communication interface includes a physical connector exposed on the housing of the first controller and not present on the housing of the partner controller.
14. The safety controller of claim 1 wherein the housing of the primary controller includes a user accessible switch defining a run and a program state and wherein a user accessible switch defining a run and program state is not included on the housing of the partner controller.
15. The safety controller of claim 1 wherein the housing of the primary controller provides a user settable a run and a program state;
wherein the transfer programs communicate the run and program state defined by the user to the partner controller; and
wherein the synchronization program executes the safety program information in the primary and partner controller according to the run and program state.
16. The safety controller of claim 1 wherein the communication interface further operates to upload safety program information to a user from the primary controller without uploading corresponding safety program information from the partner controller.
17. The safety controller of claim 1 wherein the safety program information is a set of control instructions.
18. The safety controller of claim 1 wherein the safety program information is variables used by a safety program.
16. The safety controller of claim 1 wherein the safety program information is at least one instruction causing an editing of a safety program.

17. The safety controller of claim 1 wherein the safety program information is at least one value of a variable used by a safety program.

18. The safety controller of claim 1 wherein the communication bus is a backplane having releasable electrical connectors allowing connection of the primary and partner independent controller to and from the backplane.

19. The safety controller of claim 1 wherein the communications bus is a serial communications network connecting the primary and partner controller.

20. A safety controller comprising:
a primary controller including a memory for holding program information;
a communication interface for receiving program information from a user,
the programming information including an identifier indicating whether the
programming information is a safety task;
a loader program reading program information from the communication
interface and:

(i) when the program information is a safety task, determining
whether a partner controller is in communication with the primary controller
and if a partner controller is present, loading the memory of the primary
controller with the program information and transmitting the program
information to the partner controller; and
(ii) when the program information is a not safety task, loading the
memory of only the primary controller with the program information.

21. The safety controller of claim 20 further wherein the loader program
rejects safety tasks when a partner controller is not in communication with the
primary controller.

22. A programming tool for a controller providing:
a program executable on an electronic computer to:
(i) accept program instructions from a user describing the logical
combination of input sensor data to produce output control data;

(ii) collect the program instructions into logical task;
(iii) identify the task as to one of two levels of reliability, a first level executable on a single processor and a second level requiring execution in tandem on two processors having an ability to compare execution to determine a fault in either of the two processors and to then enter a safety state;
whereby a controller receiving the tasks may automatically configure itself for the proper level of reliability or indicate a failure if that level of reliability cannot be obtained.

23. The programming tool of claim 23 wherein the program further accepts variable definitions from the user describing variables used by the program instructions, the variable definitions identifying the variables as to tasks identified to one of the two levels of reliability;

whereby variables may be properly allocated within the controller architecture for high reliability storage and modification.

24. A method of operating a safety controller having a primary and partner independent controller communicating on a communication bus comprising the steps of:

- (a) receiving safety program information from a user to the primary controller;
- (b) transferring the safety program information received by the primary controller via the communications bus to the partner controller; and
- (c) executing the safety program information on the primary and partner controller and comparing execution of the safety program information to enter a safety state when this execution differs.

25. The method of claim 24 including the step before step (a) of confirming the existence of the partner controller having the transfer and synchronization program to receive safety program information only when the confirmed partner controller is communicating with the primary controller on the communications bus.

26. The method of claim 24 wherein the safety program information executes to generate outputs to be used to control an external device and wherein step (c) compares execution of the safety program information by comparing outputs generated by the primary and partner controller executing the safety program information.

27. The method of claim 26 wherein the safety program information is executed repeatedly and wherein the comparison of the outputs is performed at the conclusion of each repeated execution prior to outputting of the outputs to the external device.

28. The method of claim 24 wherein the safety program information executes to generate values of internal variables and wherein step (c) compares execution of the safety program information by comparing values of internal variables generated by the primary and partner controller executing the safety program information.

29. The safety controller of claim 28 wherein the safety program is executed repeatedly and wherein the comparison is performed at a period greater than the repetition period.

30. The method of claim 24 wherein step (b) transfers the safety program information from the primary controller to the partner controller and receives an acknowledgement from the partner controller to the primary controller indicating that the transfer was complete and correct.

31. The safety controller of claim 30 wherein step (b) transfers the state program information in portions and receives an acknowledgement for each portion.

32. The method of claim 24 wherein the primary controller holds an identification value indicating to a user device having safety program information that the primary controller may receive safety program information and wherein the partner controller does not hold the identification value indicating to the user device

having safety program information that it may receive safety program information and wherein at step (a) the user employs the identification value to identify the primary controller.

33. The method of claim 24 wherein the communication interface also receives standard program information and wherein the safety program information holds an identification value indicating that it is part of a safety application and wherein step (b) checks for this identification value to automatically load only the safety program information received by the primary controller via the communications bus to the partner controller.

34. The method of claim 24 wherein the primary controller provides for a run and a program state and wherein step (b) communicates the run and program state to the partner controller; and step (c) executes the safety program information in the primary and partner controller according to the run and program state.

35. The method of claim 24 including step (d) of upload safety program information to a user from the primary controller without uploading corresponding safety program information from the partner controller.

36. The method of claim 24 wherein the safety program information is a set of control instructions.

37. The method of claim 24 wherein the safety program information is variables used by a safety program.

38. The method of claim 24 wherein the safety program information is at least one instruction causing an editing of a safety program.

39. The method of claim 24 wherein the safety program information is at least one value of a variable used by a safety program.

40. A method of operating a safety controller having a primary controller including a memory for holding program information comprising the steps of:

- (a) receiving program information from a user, the programming information including an identifier indicating whether the programming information is a safety task; and
- (b) determining whether a partner controller is in communication with the primary controller; and
 - (i) when a partner controller is present, loading the memory of the primary controller with the program information and transmitting the program information to the partner controller; and
 - (ii) when the program information is not safety task loading the memory of only the primary controller with the program information.

41. The method of claim 40 wherein at step (b)(i) when a partner controller is not in communication with the primary controller present, rejecting the safety tasks from the user.

42. A method of programming a safety controller having a primary and partner independent controller communicating on a communication bus comprising the steps of:

- (i) accepting program instructions from a user describing the logical combination of input sensor data to produce output control data;
- (ii) collecting the program instructions into logical task;
- (iii) identifying the task as to one of two levels of reliability a first level executable on a single processor and a second level requiring execution in tandem on two processors having an ability to compare execution to determine a fault in either of the two processors and to then enter a safety state; and
- (iv) transmitting the tasks to the safety controller so that the safety controller can automatically configure itself for the proper level of reliability or indicate a failure if that level of reliability cannot be obtained.